# Introduction to Using SQL\*PLUS and SQL Developer (12c)

## Using SQL\*Plus

* Open command prompt and enter the below command to open SQLPLUS prompt

**SQLPLUS sys as sysdba**

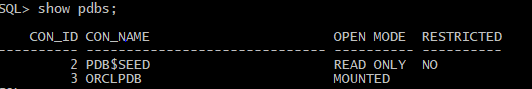
Enter the default password **Oracle\_1**.

Note: If you had chosen a different administrative password during installation, replace Oracle\_1 with the appropriate password in the command.

Now you have logged into database as sys user with sysdba permissions.

* After logging in, you have to check the pluggable databases(PDB) that are present and verify the openmode / state of those pdbs. To view that information, enter the below command.

**Show pdbs;**



In the above image, you can observe that ORCLPDB is not opened and is in mounted state.

For some, the PDB might be PDBORCL. If the PDB is in mounted state, follow the below steps.

* Before opening the database, we must change our session to that particular container. To change the session enter the below command

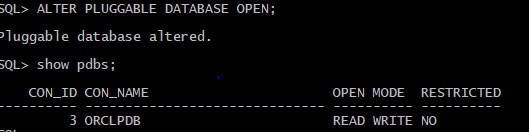
**Alter session set container=ORCLPDB;**



* Now Open the database by entering below command

**ALTER PLUGGABLE DATABASE OPEN;**

Enter the show pdbs command to check whether the open mode has changed to READ WRITE.



If your PDB is already open, you may get an error – it is fine.

**Creating a User and granting permissions:**

* To create a user, enter the below command:

**CREATE USER <username> IDENTIFIED BY <Password>;**

Replace the username and password with your own.

Example: **CREATE USER adbstudent IDENTIFIED BY adb12345;**



* Now let us grant permissions to this user. You can grant roles and privileges to the user.

To know the roles and privileges that you can grant to a user visit the below link:

<https://docs.oracle.com/cd/A97630_01/server.920/a96521/privs.htm>

Here let us grant Connect, Resource, DBA roles to the user we have created earlier.

Enter the below command to grant roles to the user:



Learn the operations that a user can perform if user is granted the above permissions.

* Check whether the above user can connect to database, enter the below command:

**Connect <Username>/<Password>@<PDBname>**



* You can enter SQL statements directly in SQL\*Plus. Be sure to start each phrase of the SQL statement on a new line, as shown here, and terminate each statement with a semicolon.

Example – type this statement to see all the tables for this user account:

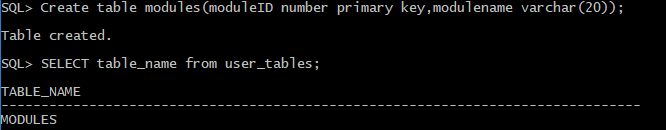
**SELECT table\_name from user\_tables;**

As there are no tables created as of now, result will be shown as no rows selected.

* Now let us create a sample table named modules which has moduleID as primary key and modulename.

**Create table modules(moduleID number primary key,modulename varchar(20));**

After the table is created, you can view tablename in user\_tables.

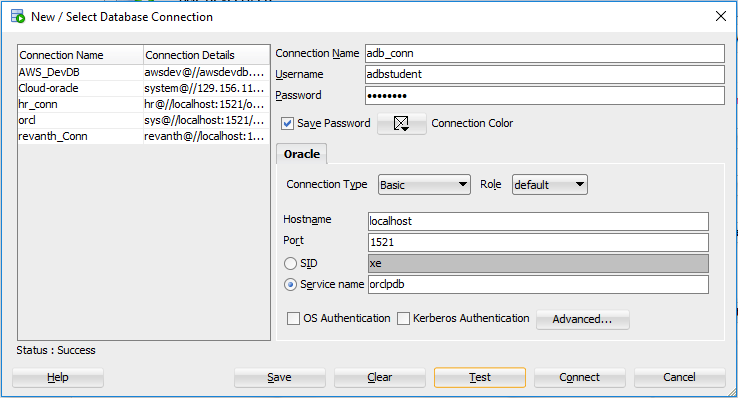


## Using SQL Developer:

You can also run scripts and execute SQL statements directly in SQL Developer.

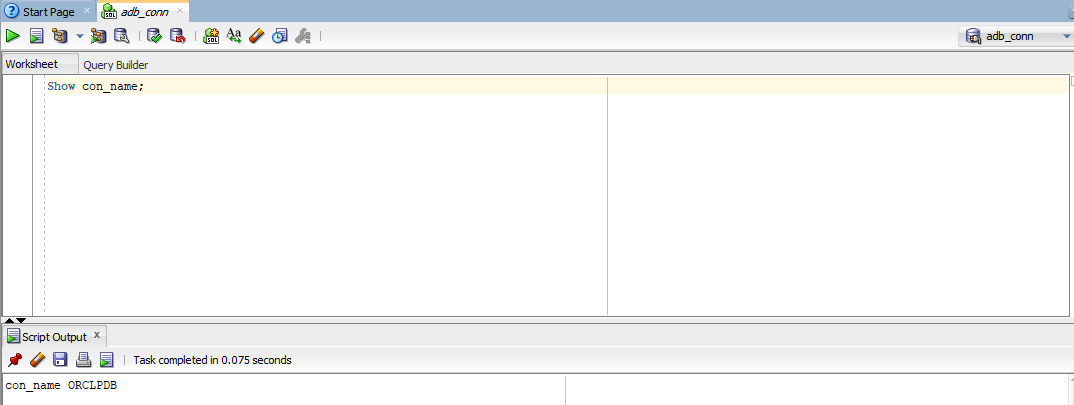
* Now, create a connection for the user to the pluggable database created earlier. Add a new connection, enter the following and click test.

If **Status** (in the lower left-hand corner) is listed as **Success**, click on **Connect**.

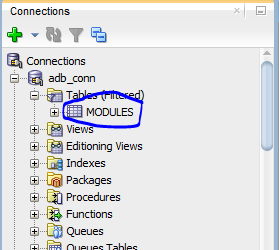


* Verify the connection name by typing the following command and running in the window:

**Show con\_name;**



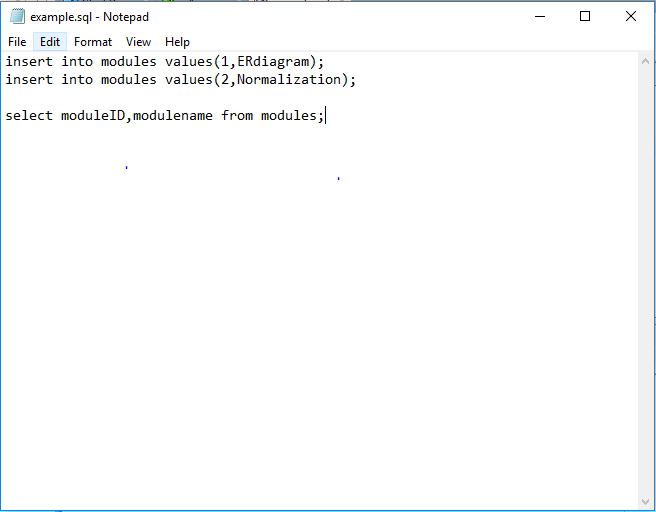
* You can view the tables that are created by the user by expanding the connection that we have created in the connection window on the left side.



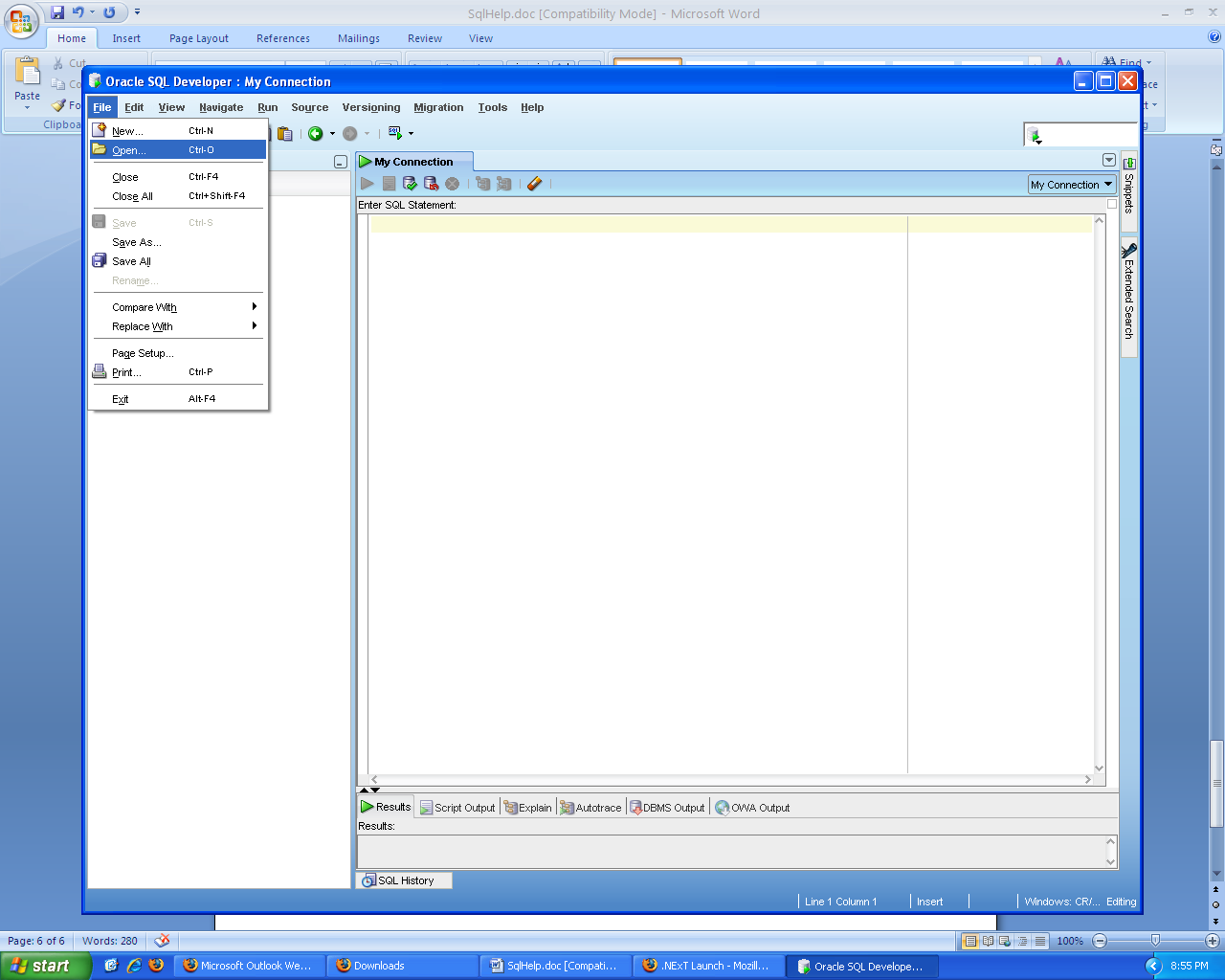
## Using Scripts:

A script is a collection of one or more SQL statements that you want to execute together.

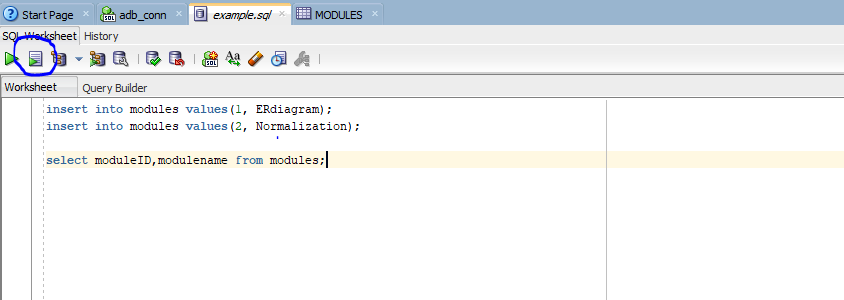
Open Notepad and enter the below statement in it and save the file with .sql extension.



After establishing a connection, click on File and select Open to open a sql script file.



Select the script you created earlier, and run it by pressing **F5**, or press the **Run Script** button.



NOTE: If you edit the script file and click on the Save icon the changes are reflected in the original script file.